Lecture

**Introduction to SQL**

Topics:

* Relational Databases
* SQL Fundamentals (CRUD)
* Primary Keys
* Foreign Keys
* SQLite3 and PostgreSQL

Relational Databases

* s
* WE will be using SQLite 3 & PostgreSQL – they are both downloaded
* Sqlite is an embedded database, meant to be used by 1 reader and 1 process accessor at a time
* Not meant for pro use – it can only support 1 query at a time, not very practical for a developer.
* Basically it means, do my thing, and till its done don’t let anyone do anything else.
* Postgresql – is much more serious, and most popular open source sql database.
  + this is the one we usually see used in development. – takes a bit more time to set up, its its own application. It runs independent of ur program, if you turn off ur program, postgres will still be running and managing many other running applications.
* SQL today and tomorrow we will use Postgres with ruby
* wednesday is ORM – object relational mapping – take a class and map each attribute into a table
* Then spend the rest of the week using Active Record – Test on friday will be on sql and active record

1 -1 relationships

* Student locker .. Each student has 1 locker so it’s a 1-1 relationship

1 – Many relationships

* 1 cohort at LHL has many students

Many – Many relationships

* In university – a given STUDENT belongs to many COURSES and all COURSES have many STUDENTS

For many-many relationships 🡪 create a join table.

SQL on computer.

CREATE TABLE artists (

id ITEGER PRIMARY KEY AUTOINCREMENT NOT NULL

name VARCHAR(50) NOT NULL

);

id integer primary key are columns in the table – it autoincrements and not null (which means it needs to have values)

commands

sqlite3 music.db – how to connect to your database

sqlite3 music.db < seed.sql 🡪 importing

sqlite3 music.db 🡪 open repl

.help 🡪 gives commands that are useful.

.tables 🡪 list tables

.schema “table” 🡪 check schema

Breakout

notes in book.

now code --- faisal created a database – coding on it now. – dvd rental database

Actor, Actor info, film\_actor(movie\_roles)